AMENDMENTS TO THE CLAIMS

1.   (Previously presented) An asynchronous programming environment, comprising:

a dynamic object storage scheme for storing a plurality of objects;

a dynamic dispatch scheme for invoking an action that belongs to one of a plurality of categories, the plurality of categories including needing one object, needing more than one object, and needing no object; and

an object recognition scheme for providing a description of each object of the plurality of objects, the description allowing a determination of whether an object described by the description fits an application programming interface.

2.   (Original) The environment of claim 1, wherein the plurality of objects as stored via the dynamic object storage scheme are accessible utilizing a recyclable locking mechanism.

3.   (Original) The environment of claim 1, wherein the plurality of objects as described via the object recognition scheme each comprise a series of tokens, each token relating to an attribute of the object.

4.   (Original) The environment of claim 1, wherein the dynamic dispatch scheme provides for execution of objects based on unpacked-into-messages events.

5.   (Previously presented) A method comprising:

storing a plurality of objects via a dynamic object storage scheme;

dispatching at least one of the plurality of objects via a dynamic dispatch scheme based on events from at least one of the plurality of objects, the dynamic dispatch scheme capable of invoking an action that belongs to one of a plurality of categories, the plurality of categories including needing one object, needing more than one object, and needing no object; and

describing each of the plurality of objects utilizing an object recognition scheme, the object recognition scheme providing a description of each object of the plurality of objects, the

description allowing a determination of whether an object described by the description fits an application programming interface.

6.   (Original) The method of claim 5, wherein storing a plurality of objects via a dynamic object storage scheme comprises accessing one of the plurality of objects utilizing a recyclable locking mechanism.

7.   (Original) The method of claim 5, wherein describing each of the plurality of objects utilizing an object recognition scheme comprises describing each of the plurality of objects as a series of tokens, each token relating to an attribute of the object.

8.   (Original) The method of claim 5, wherein dispatching at least one of the plurality of objects via a dynamic dispatch scheme comprises executing at least one of the plurality of objects based on unpacked-into-messages events.

9.   (Currently amended) A computer comprising:

a processor;

a computer-readable medium; and

an asynchronous programming environment executed by the processor from the computer-readable medium, the environment comprising:

a dynamic object storage scheme for storing a plurality of objects;

a dynamic dispatch scheme based on events from at least one of the plurality of objects, for invoking an action that belongs to one of a plurality of categories, the plurality of categories including needing one object, needing more than one object, and needing no object; and

an object recognition scheme, for providing a description of each object of the plurality of objects, the description allowing a determination of whether an object described by the description fits an application programming interface.

10.   (Original) The computer of claim 9, wherein the plurality of objects as stored via the dynamic object storage scheme are accessible utilizing a recyclable locking mechanism.

11.   (Original) The computer of claim 9, wherein the plurality of objects as described via the object recognition scheme each comprise a series of tokens, each token relating to an attribute of the object.

12.   (Original) The computer of claim 9, wherein the dynamic dispatch scheme provides for execution of objects based on unpacked-into-messages events.

13.   (Currently amended) The computer of claim 9, wherein the computer-readable medium comprises a memory.

14.   (Previously presented) A computer-readable medium having a computer program stored thereon for execution on a computer, the computer program providing an asynchronous programming environment comprising:

a dynamic object storage scheme for storing a plurality of objects;

a dynamic dispatch scheme based on events from at least one of the plurality of objects for invoking an action that belongs to one of a plurality of categories, the plurality of categories including needing one object, needing more than one object, and needing no object; and

an object recognition scheme for providing a description of each object of the plurality of objects, the description allowing a determination of whether an object described by the description fits an application programming interface.

15.   (Currently amended) The computer-readable medium of claim 14, wherein the plurality of objects as stored via the dynamic object storage scheme are accessible utilizing a recyclable locking mechanism.

16. (Currently amended) The <u>computer-readable</u> medium of claim 14, wherein the plurality of objects as described via the object recognition scheme each comprise a series of tokens, each token relating to an attribute of the object.

17. (Currently amended) The <u>computer-readable</u> medium of claim 14, wherein the dynamic dispatch scheme provides for execution of objects based on unpacked-into-messages events.

18. (Currently amended) The <u>computer-readable</u> medium of claim 14, wherein the <u>computer-readable</u> medium comprises a compact disc read only memory (CD-ROM).

19. (Currently amended) The <u>computer-readable</u> medium of claim 14, wherein the <u>computer-readable</u> medium comprises a floppy disk.

MSFT\17210AM3.DOC